

---

증명하기     $\longleftrightarrow$     프로그램 짜기

---

“논리적인 비약 없이 새로운  
사실을 확인해 가는 과정이다.”

$\longleftrightarrow$     공짜 없이 새로운 데이터를  
만들어 가는 과정이다.

“참인 사실 혹은 사실이라고  
가정한 것들로부터 시작한다.”

$\longleftrightarrow$     기초적인 데이터에서부터  
시작한다. 기초적인 데이터는  
프로그래밍 언어에서 기본적으로  
제공하는 종류의 것들이다.  
정수, 문자, 참 거짓 등.

“사실을 기반으로 해서 새로운  
사실들을 만들어 간다.”

$\longleftrightarrow$     이미 만든 데이터를 가지고  
새로운 데이터들을 만들어 간다.

“만들어 가는 과정은 근거 없는  
전너뛰기 없이, 논리적으로  
누구나 수긍하는 추론의  
정점다리를 밟고 가는  
과정만 있다.”

$\longleftrightarrow$     새 데이터를 만드는 과정은  
공짜가 없다. 사용하는  
프로그래밍 언어에서 제공하는  
프로그램 조립 방식을 써서  
만든다.

---

---

논리추론의 징검다리



프로그램 조립 방법

---

“A가 사실이고 B가 사실이면,  
A그리고B는 사실이다.”



프로그램에서 여러 데이터로 데이터  
덩치를 만드는 것에 해당한다. 특히  
두 데이터로 데이터 짝을 만드는 것에  
해당한다.

$$\frac{A \quad B}{A \wedge B}$$

A데이터가 있고 B데이터가 있으면 그  
두 개를 짝으로 모을 수 있다.

프로그래밍 언어는 두 개의 데이터로  
짝을 만드는 방법을 제공한다.

예: 아래는 괄호와 콤마로 짝을 만드는  
프로그램을 보여준다. 두 종류의  
데이터(문자열 "juliet"과 정수  
16)로 짝을 만들고 lady라고 이름  
붙이는 프로그램이다.

`lady = ("juliet", 16)`

---

“A그리고B가 사실이면,  
A가 사실이고 B가 사실이다.”



프로그램에서 데이터 덩치를 사용하는  
것에 해당한다. 특히 데이터 덩치 안에  
있는 데이터 하나를 꺼내오는 것에  
해당한다.

$$\frac{A \wedge B}{A} \quad \frac{A \wedge B}{B}$$

A데이터와 B데이터를 짝으로 묶진 데이터가 있으면 그 구성원 중 하나를 꺼내올 수 있다. 프로그래밍 언어는 그런 방법을 제공한다.

예: 앞의 프로그램 예에서 짝으로 만든 lady의 첫 번째 구성원(문자열 "juliet")을 가져오는 프로그램은 다음과 같이 적을 수 있다.

`lady.first`

두 번째 구성원(정수 16)을 가져오는 프로그램은 다음과 같을 것이다.

`lady.second`

“A가 사실이라는 가정하에 B가 사실이면, A이면 B가 사실이다.”

$$\frac{\overline{A} \vdots B}{A \Rightarrow B}$$

←→

프로그램에서 함수를 정의하는 것에 해당한다.

인자(A데이터)가 주어졌다고 가정하고 그것을 가지고 새로운 데이터(B데이터)를 만드는 프로그램이 함수 정의다. 프로그래밍 언어는 함수를 정의하는 방법을 제공한다.

예: 함수를 정의하는 아래 예를 보자.

`pair(x) = (x-1, x+1)`

인자 x로 정수(A데이터)가 들어온다고 가정하면, 그것을 가지고 정수 짝(B데이터)을 만드는 함수를 프로그램한 것이다.

“ $A$ 이면  $B$ 가 사실이고  $A$ 가 사실이면,  $B$ 가 사실이다.”

$$\frac{A \Rightarrow B \quad A}{B}$$

프로그램에서 함수를 사용하는 것에 해당한다.

$A$ 데이터가 주어지면  $B$ 데이터를 만드는 함수가 있다고 하자. 이 함수에  $A$ 데이터의 값을 주면 그 함수는  $B$ 데이터 값을 만들게 된다.

프로그래밍 언어는 함수를

적용하는(데이터를 함수에 넘겨서 함수를 작동시키는) 방법을 제공한다.

예: 아래 프로그램은 함수 `pair`에 정수 3을 주어서 그 함수를 작동시키는 프로그램이다.

`pair(3)`

“ $A$ 가 사실이면, 임의의  $B$ 에 대해서  $A$ 또는 $B$ 와  $B$ 또는 $A$ 가 사실이다.”

$$\frac{A}{A \vee B} \quad \frac{A}{B \vee A}$$

프로그램에서 데이터를 뭉뚱그리는 것에 해당한다.

$A$ 데이터가 있으면 그 데이터를  $A$ 또는 $B$  데이터라고 뭉뚱그리는 것이다. 예를 들어 프로그램에서 사과를 만들고 과일(사과 또는 배)이라고 뭉뚱그리는 것이다.

그래서  $A$ 또는 $B$  데이터(과일) 속에는  $A$ 데이터(사과) 혹은  $B$ 데이터(배) 하나만을 갖고 있다.

프로그래밍 언어는 이런 뭉뚱그리는 방법을 제공한다.

“ $A$ 또는 $B$ 가 사실이고  $A$ 라는 가정하에  $C$ 가 사실이고,  $B$ 라는 가정하에서도  $C$ 가 사실이면,

프로그램에서 뭉뚱그린 데이터를 사용하는 것에 해당한다.  $A$ 또는 $B$  데이터라고 뭉뚱그린 것이 있다. 그 뭉뚱그린 것이  $A$ 데이터일

그러면  $C$ 는 사실이다.”

$$\frac{\overline{A} \quad \overline{B} \quad \vdots \quad \vdots \quad A \vee B \quad C \quad C}{C}$$

경우  $C$ 데이터를 만드는 프로그램이 있고,  $B$ 데이터일 경우  $C$ 데이터를 만드는 프로그램이 있다면, 이 프로그램을 가지고  $A$ 또는  $B$  데이터를 받아서 각 경우에 따라  $C$ 데이터를 만드는 프로그램을 만들 수 있다. 프로그래밍 언어는 이런 프로그램을 만드는 방법을 제공한다.

예: 아래는 사과나 배를 과일로 뭉뚱그린 데이터  $x$ 를 받아서 각 경우마다 잼을 만드는 프로그램이다.

```
case x of Apple => makeAppleJam(x)
        | Pear  => makePearJam(x)
```

“가정은 언제나 가능하고,  
가정한 것은 사실이라고  
생각하고 증명한다.”

$\overline{A}$

↔

프로그램에서 함수의 인자를 사용하는 것에 해당한다. 어떤 데이터가 있다고 가정하고 프로그램을 짤 수 있다. 그런 데이터를 인자로 받는 함수를 정의할 때 그렇다. 프로그래밍 언어는 함수의 인자에 이름을 붙이고 그 이름을 함수 내부에서 자유롭게 사용할 수 있게 한다.